Austin Bennett

15-112

TP2: Design Proposal

Design Proposal for Gay Simulator

Project Proposal:

      The project will be called *Gay Simulator*, and it will be a "Bullet Hell", a kind of first person shooter. The player will play as a gay person attempting to spread tolerance (in the form of an LGBTQIA+ Pride Flag) to the various enemies and will receive points for doing so. The player moves with the standard WASD keys and shoot by clicking the mouse at the desired target. Throughout the game, the enemies will try to spread intolerance; to help against this, the player can pick up powerups. These powerups include various gay icons, a pride parade, and ecstasy. The game ends when a player's health bar reaches zero or when the player chooses to quit the game. The goal of the game is to provide the player with a fun game while teaching LGBTQIA+ tolerance.

Competitive Analysis:

      Gay Simulator was inspired by many famous first person shooters, in particular, "Bullet Hell" first person shooters. While being relatively modest for a "Bullet Hell", *Gay Simulator* takes on this style of game play to show the overwhelming everyday difficulties that a gay person has to face. In terms of gameplay, the game is similar to *Galaga* in structure; however, the player is automatically moved over the screen, the flags shoot in the direction of the mouse click from the center of the player, and there will be many more projectiles. The game also takes elements from many LGBTQIA+ resources in order to teach tolerance to a large audience; however, the game attempts to show problems LGBTQIA+ people face in a much lighter way to help teach the user tolerance while still being a fun game. Overall, *Gay Simulator* combines elements from first person shooters and LGBTQIA+ resources in an attempt to teach a wide variety of people about tolerance in a fun way.

Structural Plan:

      The game will be comprised of a game board, enemies, powerups, and the player. Those four elements are broken down into the following structures:

Classes: Barrier (a board structure), Player (the player's character), Chad (an enemy), Mississippi (an enemy structure), Westboro Baptist Church (an enemy structure), Westboro Baptist Church Member (an enemy), and Ecstasy (a powerup).

Files: There will be imported GIFs for the sprites on the board for the enemies and the player.

Functions: summonRuPaul (a gay icon powerup), summonAlanTuring (a gay icon powerup), summonLadyGaga (a gay icon powerup), Move (Moves the objects that can move), isCollision (Checks for a collision between two objects), onTimerFired (Executes all actions that are updated on timer fired), drawBoard (Which draws the board), makeEnemy(Makes the selected enemy), hitWithFlag/hitWithProj(Checks if enemy/player was hit with a flag/enemy projectile), Draw (Draws An Object), start/gameOver (Draws the Game Over/Starting Screens), init (Initializes important values), mousePressed(Executes various tasks on a mouse press), keyPressed(Executes various tasks on a key press), timerFired (Executes various tasks on a timer fired), redrawAll (Draws the board), and many helper functions (and probably a few more functions if structure is changed).

TP2 Update: Re-did the shooting and following to be based more on player location, added extra building to the drawboard function, Lady Gaga powerup now also has Britney (bc she is just as much of a queen), added movement/interaction with bills (slow player down), enemies (-50 health) and NoHomo (-25 health), implemented powerups, basically drew everything, and many small tasks.

TP3 Update: Graphics added, levels added, start screen added, instruction screen added, end screen added, gifs added for powerups, collision bugs fixed, removed ecstasy powerup (toxic gameplay), and dead code deleted.

Algorithmic Plan:
    While the game doesn't appear to have one big, complex problem, I believe that there are three large tasks to tackle.

Player Shooting/Movement - This involves the shooting of the rainbow flags, and how the player moves around the board. Luckily, I was able to solve this after hackTP and some further work on the project.

Enemy Shooting/Movement - This involves having the enemies move toward the player and all different kinds of projectiles thrown at the player. I plan on using some functions from math, some simple trig, and some trial and error to solve these problems.

Efficiency - This involves making sure that all parts of the game are moving properly (as there will be so many projectiles that the game could slow down). I plan on solving this by using sets

and getting rid of all unnecessary values (something I have already done with the rainbow flag projectiles.)

Timeline:
Accomplished: Player Shooting/Movement, Drawing gameboard, basic enemy organization, getting enemies to move toward player, enemy attacks, scoring, level elevation, powerups, and game over.
Wednesday → Importing graphics/Animation, instructions (Not sure what to do exactly ask TP2 meeting)
Thursday → Importing graphics/Animation and creating different game modes (if time)
Friday → Bug fixes, finishing touches

Version Control:
In order to ensure the safety of my project, I will be copy-pasting all code into a google drive file called Gay Simulator Backup. (Picture attached below)
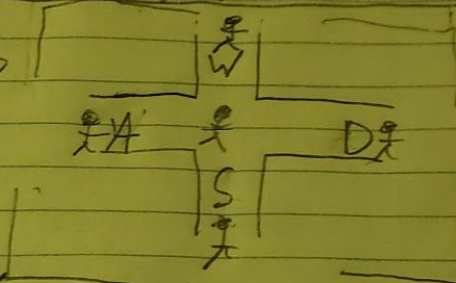
```
#################################################################
# 15-112-n18 hw15
# Your Name: George Austin Bennett
# Your Andrew ID: gbennett
# Your Section: A
#################################################################
import math
#################################################################
# Classes:
#################################################################

class movingObject (object):
    def __init__(self, cx, cy):
        self.cx = cx
        self.cy = cy
        self.width = 50
        self.height = 50

    def isCollision(self, other):
        objectPoints = [(self.cx - self.width//2, self.cy - self.height//2),
                        (self.cx + self.width//2, self.cy - self.height//2),
                        (self.cx - self.width//2, self.cy + self.height//2),
                        (self.cx + self.width//2, self.cy + self.height//2),
                        (self.cx, self.cy)]
        for oPoint in objectPoints:
```

Storyboard:

GAY SIMULATOR
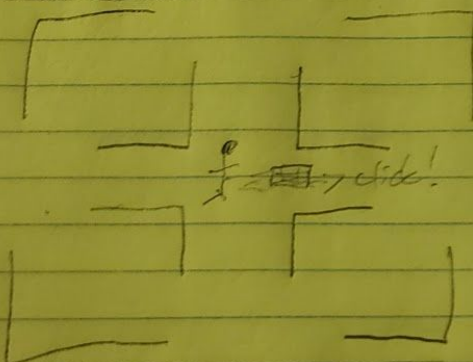
① Title Screen/Instructions

Welcome to Gay Simulator!

Rules:
1. Click the mouse at where you fire
2. Move with WASD
3. Eliminate all homophobes
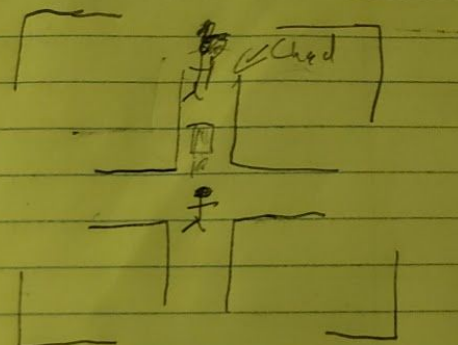4. Try not to and hit any projectiles
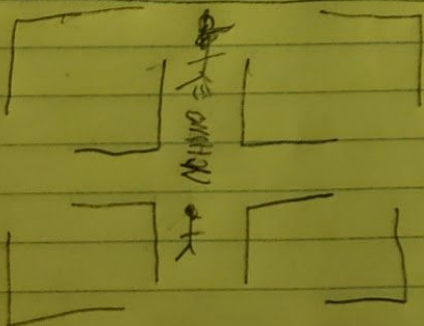5. HAVE FUN!

② Player moves with WASD

③ Player shoots rainbow with click

by click!

④ Player can shoot enemies

Chad

⑤ Enemies shoot back!

⑥ Mississippi and WBC spawns bills and weapons
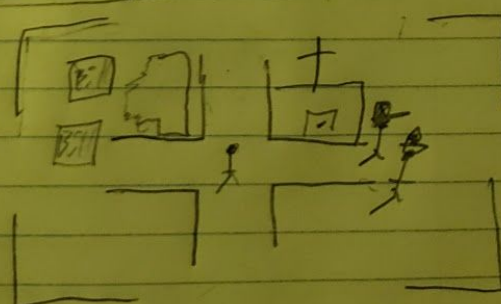
⑦ Gay Icons & Power Ups Help!
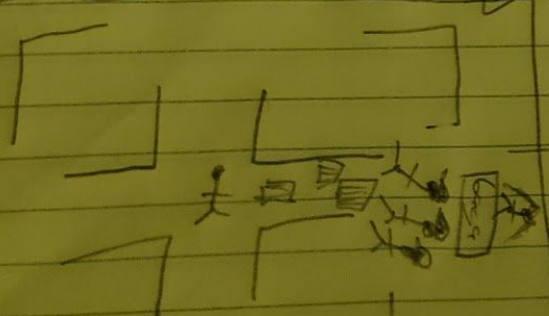
⑧ Game only when player runs out of health

Game Over!

Score: #

Press "Z" to play again!